

メディアプログラミング演習—第12回（第6テーマ1日目）—

テーマ6：3Dグラフィックスの扱い

コンピュータグラフィックス（以下、CGと書く）の技術は、幅広い分野に応用されている。CGに関する技術には大別して、「コンピュータ内に形状を表現する技術：モデリング（Modeling）」、および、「画像として形状を表現する技術：レンダリング（Rendering）」の2つがある。一般のCGソフトには各々特徴とするところはあるが、ほとんどの場合、これらの2つの機能が実装されている。第6テーマとして、3次元の立体の構成（モデリング）とその表示（レンダリング）を扱う。

processingの3Dモードにおける座標系は、図12-1に示す様に、ウインドウ左上が原点、右方向がx軸、下方向がy軸、画面手前方向がz軸である。

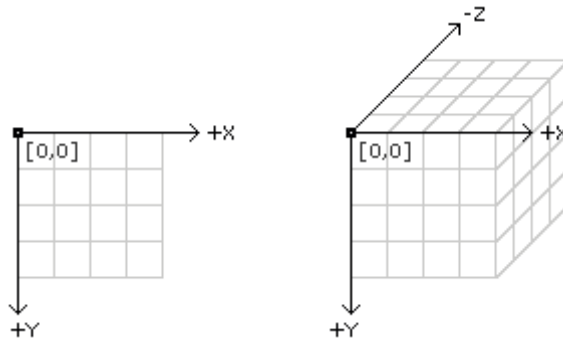


図12-1 2Dおよび3Dの座標系

図12-2は、空間内の平面を描くプログラムの一部である。空間内の平面は、その頂点を順に指定することにより描かれる。頂点列はbeginShape()からendShape()の間に指定する。

beginShapeの括弧内は、その点列をつなげてどのような形を形成するかを指定する。TRIANGLESは、3頂点を指定して3辺形を、QUADSは4点を指定し4辺形を形作る。他の指定方法の説明は省略する。

空間平面の描画

プログラムmd-simple-shapeは、この部分を含み、表示全体のプログラムである。

文size(800, 800, P3D)は扱う座標系が3次元であることを指定する。noFill()および

stroke(255)は、後に変更するが、図形を「塗り潰しせず」、「輪郭を白」で描くことを指定する。最後の関数keyPressed()は、矢印キーにより回転角度を操作する関数であり、実際の回転の指定は関数draw()にある。

```
void draw3Dspace() {  
  // 空間内の平面  
  beginShape(QUADS);  
  vertex(0, 0, 0);  
  vertex(0, 1, 0);  
  vertex(1, 1, 0);  
  vertex(1, 0, 0);  
  endShape();  
}
```

図12-2 サンプルプログラム

n 角錐の描画

xy 平面を底面 (半径 1) とし高さ 1 の角錐の描画モデルを図 12-3 に示す. n 角錐は, 周りの n 個の三辺形, および, 底辺から成り立っている

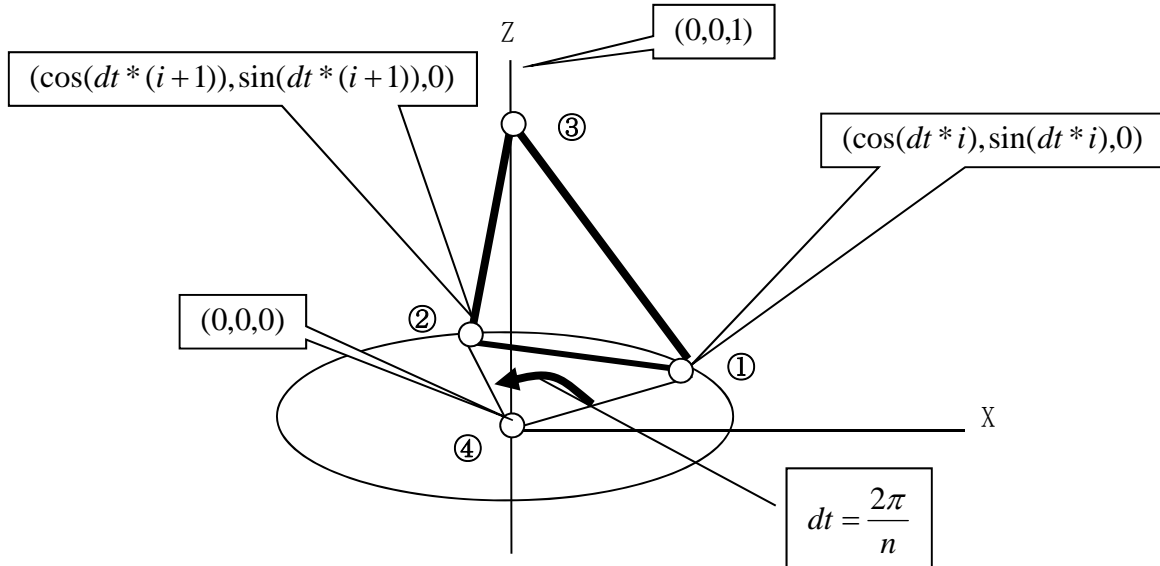


図12-3. n角錐の各頂点の座標計算法

図 12-3 に示す様に, 周りの i 番目の三辺形の三頂点①②③は, 以下の通りとなる.

$$(\cos(dt * i), \sin(dt * i), 0), (\cos(dt * (i + 1)), \sin(dt * (i + 1)), 0), (0, 0, 1)$$

また, 底辺は n 辺形であるので, 同様に n 個の三辺形で表現すると, i 番目三辺形の頂点①②④は, 以下となる.

$$(\cos(dt * i), \sin(dt * i), 0), (\cos(dt * (i + 1)), \sin(dt * (i + 1)), 0), (0, 0, 0)$$

図形描画の中心部を図 4 に示す. ここで, 座標値は配列で表現し, 要素[0]に x 座標値, 要素[1]に y 座標値, 要素[2]に z 座標値を格納することとする.

演習 6-1 n 角錐のワイヤフレームを描くプログラムの概形が `md-cone-wf` である. 関数 `drawCone()` を完成させなさい. 関数 `drawCone()` 中, 変数 n は分割数 (36 になっている), tdi は前述の $dt*i$ を, $tdip1$ は同様に $dt*(i+1)$ を表す変数である. 図 12-4 において, ①の座標値を求める部分は `P1[0]=sin(tdi); P1[1]=cos(tdi); P1[2]=0;` となる.

演習 6-2 n 角柱 (底辺中心原点, 半径 1, 高さ 1) を描く関数 `drawcylinder()` を完成させ, 36 角柱を描きなさい. ただし, 側面は 4 辺形, 両底面共 3 辺形で表現しなさい.

```
Pi[0,1,2] <- i 番目の座標値
beginShape(TRIANGLES);
    vertex(P1[0],P1[1],P1[2]);
    vertex(P2[0],P2[1],P2[2]);
    vertex(P3[0],P3[1],P3[2]);
endShape();
beginShape(TRIANGLES);
    vertex(P2[0],P2[1],P2[2]);
    vertex(P1[0],P1[1],P1[2]);
    vertex(P4[0],P4[1],P4[2]);
endShape();
```

図 12-4 n 角錐の描画